(19)日本国特許庁（ＪＰ）　(12)公表特許公報（Ａ）　(11)特許出願公表番号
特表2003－513349
（P2003－513349A）
(43)公表日　平成15年４月８日(2003.4.8)

(51)Int.Cl.[7]	識別記号	ＦＩ	ﾃｰﾏｺｰﾄﾞ(参考)
Ｇ０６Ｆ　15/00	３１０	Ｇ０６Ｆ　15/00	３１０Ｐ　５Ｂ０８５
	３３０		３３０Ａ

審査請求　未請求　予備審査請求　有　（全 37 頁）

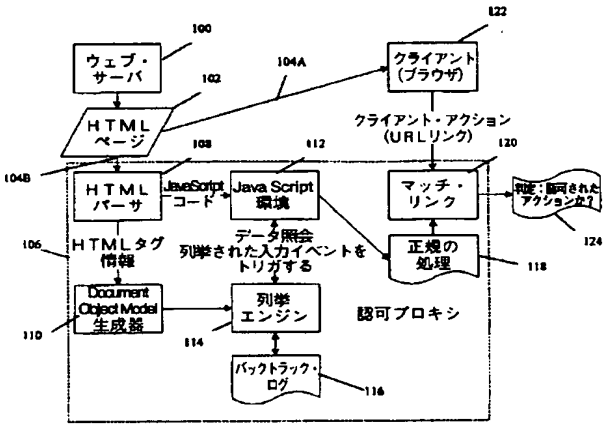
(21)出願番号　特願2001－533487(P2001－533487)
(86)(22)出願日　平成12年10月25日(2000.10.25)
(85)翻訳文提出日　平成14年４月25日(2002.4.25)
(86)国際出願番号　ＰＣＴ／ＩＬ００／００６７７
(87)国際公開番号　ＷＯ０１／０３１４１５
(87)国際公開日　平成13年５月３日(2001.5.3)
(31)優先権主張番号　６０／１６１，４７３
(32)優先日　平成11年10月25日(1999.10.25)
(33)優先権主張国　米国（ＵＳ）

(71)出願人　サンクタム、リミテッド
イスラエル国、ヘルズリヤ、サピア　ストリート　１、ピー．オー．ボックス　12047
(72)発明者　モラン、タル
イスラエル国　テル　アビブ、ホフェイン　ストリート　10
(72)発明者　エル　－　ハナニ、ユヴァル
イスラエル国　テル　アビブ、ラサル　ストリート　10
(74)代理人　弁理士　浅村　皓　（外３名）

最終頁に続く

(54)【発明の名称】　クライアントの要求を確認する方法およびシステム

(57)【要約】
クライアントとデータ処理システムのサーバとの間で送信された要求されたアクションの実行を認可するシステムおよび方法が提示されている。該方法は、アクションのセットを含んでいるメッセージを受信し、そのアクションのセットの実行をシミュレーションすることを含む。許容可能なアクションと、シミュレーションしたアクションへのユーザが定義可能な入力とを表すリストが定義される。次いで、許容可能なアクションと、許容可能なアクションへのユーザが定義可能な入力は、ユーザが要求したアクションおよび入力と比較される。ユーザが要求したアクションおよび入力内の要素が、許容可能なアクションと入力のリストに含まれているとき、ユーザが要求したアクションおよび入力は、実行を認可される。



【特許請求の範囲】

【請求項１】　クライアントとデータ処理システムのサーバとの間で送信された要求されたアクションの実行を認可する方法であって、

アクションのセットを含んでいる第１メッセージと、ユーザが要求したアクションおよび入力を含んでいる第２メッセージを受信することと、

アクションのセットの実行をシミュレーションし、許容可能なアクションと許容可能なアクションへのユーザが定義可能な入力とのリストとを構築することと、

許容可能なアクションとユーザが定義可能な入力とのリストを、ユーザが要求したアクションおよび入力と比較することとを備え、

許容可能なアクションとユーザが定義可能な入力とのリストが、ユーザが要求したアクションおよび入力を含み、ユーザが要求したアクションの実行を認可する方法。

【請求項２】　シミュレーションのステップが、クライアントにおけるアクションのセットの実行から得られるすべての可能なアクションと、可能なアクションへの入力とを識別することを備える、請求項１に記載の方法。

【請求項３】　シミュレーションのステップが、アクションのセット内にある各コマンド、フィールド、ユーザが選択可能な入力オプション、およびＨＴＴＰ要求を呼び出し、およびトリガすることを備える、請求項１に記載の方法。

【請求項４】　ユーザが要求したアクションおよび入力が、クライアントにおける第１メッセージの受信に応答して実施されたユーザ・セッション中に提供されたアクションと入力とを含む、請求項１に記載の方法。

【請求項５】　シミュレーションのステップ中に、データ値のエントリを要求する入力制御を検出し、そのデータ値を表すために、固有の配置ホルダを割り当てることと、

比較のステップ中に、固有の配置ホルダのパターンを、ユーザから受信した入力に整合させることとを備える、請求項１に記載の方法。

【請求項６】　シミュレーションのステップが、

複数の事前に定義されたデータ値の１つを選択することを要求する入力制御を

検出することと、

複数の事前に定義されたデータ値の各々が選択され、かつリストされるまで、複数の事前に定義されたデータ値１つを対話式に選択し、アクションのセットのシミュレーションを続行し、選択された１つのデータ値を有する許容可能なアクションとユーザが定義可能な入力とのリストを構築することとを備える、請求項１に記載の方法。

【請求項７】　シミュレーションのステップの前に、クライアントにおいてアクションのセットの実行を追跡することと、

シミュレーションのステップ中に、ユーザが選択可能な入力に応答して、追跡の結果を提供することとを備える、請求項１に記載の方法。

【請求項８】　シミュレーションのステップの前に、

第１メッセージのアクションのセット内にあるアクションを識別することと、

識別されたアクションへの入力を追跡するために、アクションで第１メッセージを補足することと、

補足された第１メッセージをクライアントに送信することと、

シミュレーションのステップ中に、識別されたアクションへのユーザが選択可能な入力が要求された際に、追跡の結果を提供することとを備える、請求項１に記載の方法。

【請求項９】　追跡の結果が、第２メッセージ内に含まれている、請求項８に記載の方法。

【請求項１０】　追跡の結果が、第２メッセージを受信する前に受信された第３メッセージに含まれている、請求項８に記載の方法。

【請求項１１】　クライアントとデータ処理システムのサーバとの間で結合されたセキュリティ・ゲートウェイであって、

前記クライアントとサーバとの間の送信を評価し、かつ、各送信内に含まれている情報内容とアプリケーション・プログラミング論理を識別するための評価装置と、

前記送信の前記アプリケーション・プログラミング論理を実行する処理環境をシミュレーションするシミュレータであって、事象をトリガし、かつ、前記アプ

リケーション・プログラミング論理へのユーザが定義可能な入力を識別するための列挙エンジンを含み、許容可能なアクションと、前記アクションへのユーザが定義可能な入力値とのリストを提供するシミュレータと、

ユーザが要求したアクションおよび入力を含んでいる送信を受信し、前記ユーザが要求したアクションおよび入力を、前記許容可能なアクションとユーザが定義可能な入力値とのリストと比較し、前記許容可能なアクションと入力値とのリスト内にあるユーザが要求したアクションおよび入力を有する送信を前記セキュリティ・ゲートウェイを通して渡すフィルタとを備える、セキュリティ・ゲートウェイ。

【請求項１２】　前記許容可能なアクションと入力値とのリストを記憶するために、前記シミュレータと前記フィルタによってアクセス可能なデータ・ストアを備える、請求項１１に記載のセキュリティ・ゲートウェイ。

【請求項１３】　前記シミュレータが、データ値のエントリを要求する入力制御を検出し、かつ、前記データ値を表すために、固有の配置ホルダを割り当てるための検出器を備え、前記フィルタが、前記固有の配置ホルダのパターンを、前記ユーザから受信した前記入力に整合させるための手段を備える、請求項１１に記載のセキュリティ・ゲートウェイ。

【請求項１４】　前記シミュレータが、

複数の事前に定義されたデータ値の１つを選択することを要求する入力制御を検出する検出器と、

前記複数の事前に定義されたデータ値の各々が選択され、かつリストされるまで、前記複数の事前に定義されたデータ値の１つを対話式に選択し、前記アプリケーション・プログラミング論理のシミュレーションを続行し、前記選択された１つのデータ値を有する許容可能なアクションとユーザが定義可能な入力とのリストを構築する手段とを備える、請求項１１に記載のセキュリティ・ゲートウェイ。

【請求項１５】　前記評価装置が、前記アプリケーション・プログラミング論理内において、関心のあるアクションを識別し、かつ、クライアントにおいて受信した前記アクションへの入力を追跡するための手段を備え、前記シミュレー

タが、前記追跡の結果を受信し、かつ、前記識別されたアクションへのユーザが定義可能な入力が、前記シミュレーション内において実施される際に、前記追跡の結果を提供するための手段を備える、請求項１１に記載のセキュリティ・ゲートウェイ。

【請求項１６】　クライアントからクライアント／サーバ・データ処理システムに送信された要求されたアクションの実行を認可する方法であって、クライアントとサーバの間に結合されたゲートウェイによって実施され、

サーバから、アクションのセットを含んでいるドキュメントを受信することと、

アクションのセットの実行をシミュレーションし、許容可能なアクションと、許容可能なアクションへのユーザの定義可能な入力とのリストを構築することと、

クライアントから、ユーザが要求したアクションおよび入力を含んでいるメッセージを受信することと、

許容可能なアクションとユーザが定義可能な入力とのリストを、ユーザが要求したアクションおよび入力と比較することとを備え、

許容可能なアクションとユーザが定義可能な入力とのリストが、ユーザが要求したアクションおよび入力を含み、ユーザが要求したアクションおよび入力を、実行のためにサーバに送信する方法。

【請求項１７】　ゲートウェイにおいて、許容可能なアクションとユーザが定義可能な入力とのリストを記憶することを備える、請求項１６に記載の方法。

【発明の詳細な説明】

【０００１】

（著作権の注意）

本出願ドキュメントの部分は、著作権保護を受ける内容を含む。著作権の所有者は、特許事務所または商標事務所の特許ファイルまたは記録にあるように、誰もが出願ドキュメントまたは出願の開示をファクシミリで再生することに異存はないが、そうでない場合、すべての著作権を保有する。

【０００２】

（関連する出願）

本出願は、以下の同時係属中の出願に関連する。

・１９９９年１０月２５日に出願された、弁理士整理番号第３２６９／８号であるＭＥＴＨＯＤ　ＡＮＤ　ＳＹＳＴＥＭ　ＦＯＲ　ＶＥＲＩＦＹＩＮＧ　ＡＣＬＩＥＮＴ　ＲＥＱＵＥＳＴという名称の米国仮特許出願第６０／１６１，４７３号

・１９９９年７月１日に出願された、弁理士整理番号第３２６９／６号であるＭＥＴＨＯＤ　ＡＮＤ　ＳＹＳＴＥＭ　ＦＯＲ　ＥＸＴＲＡＣＴＩＮＧ　ＡＰＰＬＩＣＡＴＩＯＮ　ＰＲＯＴＯＣＯＬ　ＣＨＡＲＡＣＴＥＲＩＳＴＩＣＳという名称の米国特許出願番号第０９／３４５，９２０号

・１９９８年９月９日に出願された、弁理士整理番号第３２６９／３号であるＭＥＴＨＯＤ　ＡＮＤ　ＳＹＳＴＥＭ　ＦＯＲ　ＰＲＯＴＥＣＴＩＮＧ　ＯＰＥＲＡＴＩＯＮＳ　ＯＦ　ＴＲＵＳＴＥＤ　ＩＮＴＥＲＮＡＬ　ＮＥＴＷＯＲＫＳという名称の米国特許出願番号第０９／１４９，９１１号

・１９９８年９月９日に出願された、弁理士整理番号第３２６９／４号であるＭＥＴＨＯＤ　ＡＮＤ　ＳＹＳＴＥＭ　ＦＯＲ　ＭＡＩＴＡＩＮＩＮＧ　ＲＥＳＴＲＩＣＴＥＤ　ＯＰＥＲＡＴＩＮＧ　ＥＮＶＩＲＯＮＭＥＮＴＳ　ＦＯＲ　ＡＰＰＬＩＣＡＴＩＯＮ　ＰＲＯＧＲＡＭＳ　ＯＲ　ＯＰＥＲＡＴＩＮＧ　ＳＹＳＴＥＭＳという名称の米国特許出願番号第０９／１５０，１１２号

・ＰＣＴ出願ＰＣＴ／ＩＬ９８／３２６９／４

・ＰＣＴ出願ＰＣＴ／ＩＬ９８／００４４３

・ＰＣＴ出願ＰＣＴ／ＩＬ００／００３７８

これらの出願の開示は、参照することによって、完全に本明細書に組み込まれている。

【０００３】

（発明の背景）

本発明は、一般に、プライバシおよびセキュリティのシステムに関し、より詳細には、データ処理システム内において、クライアント装置とホスト装置との間で送信された要求、データ、情報内容、アプリケーション、および他の情報を認可する方法およびシステムに関する。

【０００４】

（従来の技術）

クライアント／サーバ・データ処理システムでは、いくつかのパーソナル・コンピュータ、ワーク・ステーション、携帯式および／または手持ち式装置など（「クライアント」）は、１つまたは複数のホスト・コンピュータ（「サーバ」）と連結され、かつ通信している。サーバは、ネットワーク上でクライアントによって共有されている情報および／またはアプリケーション・プログラムに対するクライアントからの要求を処理する。ますます、クライアント／サーバ・ネットワークは、例えば、それ自体がワールド・ワイド・ネットワークの一部である、すなわち、インターネットのワールド・ワイド・ウェブの一部（「ウェブ」）であることが可能であるイントラネットおよびエキストラネットなど、より広範な「ネットワークのネットワーク」を形成するために、連結されるようになってきた。ネットワークを連結することにより、クライアントは、ネットワークにわたって、リソース（例えば、情報およびアプリケーション・プログラム）を供給することが可能になる。

【０００５】

潜在的にワールド・ワイド・ネットワーク上で、共有されている情報およびアプリケーション・プログラムの利用可能性が増大するにつれ、個々のクライアント／サーバ・ネットワークの各々の脆弱性が増大している。例えば、ネットワークの１つに記憶されている所有権を主張できる情報およびアプリケーション・プ

ログラムを検索および／または損傷しようとする不謹慎な個人は、所有権を主張できる情報およびアプリケーション・プログラムにアクセスして、無認可の方式で、それを使用する可能性がある。そのような無認可の使用を防止する努力では、多くのネットワークは、「ファイアウォール」を通して、他のネットワークに接続されている。従来のファイアウォールは、内部ネットワーク・リソース（例えば、特有のウェブ・サーバまたはフォルダ）へのアクセス制御に対処し、ネットワークの部分へのアクセスを制限し、および、それに記憶されている所有権を主張できる情報およびアプリケーション・プログラムに対する無認可の検索または損傷を防止するように設計されたハードウェアおよび／またはソフトウェア・システムを含む。

【０００６】

しかし、多くの従来のファイアウォール・システムは、アプリケーション・レベルでは認可に対処せず、認可されたクライアントであると偽った不謹慎な個人によって機能しなくなる可能性がある。例えば、多くのウェブ・アプリケーションは、アプリケーションのユーザは、実際に彼／彼女のブラウザ上でアプリケーションのモバイル・エージェントを実行していると想定している。しかし、悪意のあるユーザは規格のウェブ・ブラウザ・ソフトウェアを使用せずにウェブ・サーバに接続することができ、したがって、そのユーザはブラウザ側で強化することが可能である制限になんら束縛されず、悪意のあるユーザは、標準的なクライアントであると偽って、ウェブ・サーバに破壊的なまたは捏造したデータを送信することができる。

【０００７】

共通して譲受された米国特許出願第０９／３４５，９２０号には、標準的なＨＴＭＬドキュメントのユーザからの要求を確認するための解決法が記載されている。その解決法は、ＨＴＭＬドキュメントの内容（「認可されたアクション」）に基づいて、ブラウザ・ソフトウェアが取ることが可能であるアクションのセットまたはパターン（ＨＴＴＰ要求）を抽出することに基づいている。次いで、この認可されたアクションのセットは、クライアントのアプリケーションによって送信された要求に対して整合される。ユーザが、規格のブラウザの１つを使用し

ていない場合であっても、アクションの法的または認可されたセット内からの要求のみが、ウェブ・サーバに渡されることになる。

【０００８】

以上の内容を考慮して、本発明の発明者は、上述した確認技術を、ウェブ・サーバの代わりにクライアント・システム上で実行する論理（例えば、ＨＴＭＬページに埋め込まれたＪａｖａＳｃｒｉｐｔ（ＴＭ）プログラム）に拡張する必要性を認識した。特に、本発明者は、外部データとその上で起きている事象を入手および確認するために、クライアント・サイドの論理の実行をシミュレーションする必要性を認識した。

【０００９】

（発明の簡単な概要）

本発明の目的は、サーバ・システムの代わりに、クライアント・システム上で実行する論理のための確認技術を提供することである。

【００１０】

本発明の他の目的は、クライアント・サイドの論理（例えば、ＨＴＭＬページに埋め込まれたＪａｖａＳｃｒｉｐｔプログラム）のための確認技術を提供することである。この技術は、外部データとそこで起きている事象を入手および確認するために、論理の実行をシミュレーションする。

【００１１】

本発明のさらに他の目的は、外部データと事象とに対する認可された要求のみが、クライアントから保護されたサーバに渡されるように、論理の実行をシミュレーションするクライアント・サイドの論理のための確認技術を提供することである。

【００１２】

本発明の他の目的および利点は、図面とそれに続く記載を考慮することから、より明らかになるであろう。

【００１３】

以上および他の問題点は、克服され、目的は、システムおよび方法が、クライアントとデータ処理システムのサーバとの間で送信された要求されたアクション

の実行を認可するために提示されている、本発明の実施形態による方法および装置によって実現される。該方法は、アクションのセットまたはプログラム（例えば、ＨＴＭＬページに埋め込まれたＪａｖａＳｃｒｉｐｔプログラム）を含んでいるメッセージを受信することと、アクションのセットの実行をシミュレーションすることとを含む。リストは、実行されたアクションと、実行されたアクションへのユーザが定義可能な入力とを表すように定義される。次いで、実行されたアクションと、実行されたアクションへのユーザが定義可能な入力のリストとは、許容可能なアクションと入力のリストと比較される。実行リスト内の要素が許容可能なアクションと入力のリストに含まれているとき、メッセージとアクションの含まれていたセットは認可される。

【００１４】

本発明は、例示的であり、限定的でないことを意図している添付の図面の図に示されている。図では、同じ参照符号は、同じまたは対応する部分を指すことを意図している。

【００１５】

（好ましい実施形態の詳細な説明）

図１Ａは、本発明の一実施形態により構成されかつ動作するクライアント／サーバ・データ処理ネットワーク１０を示す。データ処理ネットワーク１０は、１つまたは複数のホスト／サーバ・コンピュータと連結され、かつ通信している、パーソナル・コンピュータ、ワーク・ステーション、携帯式および／または手持ち式装置など、複数のクライアント・システムを含む。図示を明瞭にするために、複数のクライアントは、クライアント・システム１２によって表し、１つまたは複数のサーバは、サーバ１４によって表している。クライアント１２とサーバ１４は、例えばインターネットのワールド・ワイド・ウェブの部分（「ウェブ」）、イントラネット、エキストラネット、または他の個人的なネットワークなど、通信ネットワーク１６に有線接続または無線接続を介して結合された、遠隔装置および／または局所装置とすることが可能であることを理解されたい。

【００１６】

本発明によれば、認可プロキシ・システム１８は、クライアント１２とサーバ

１４の間で結合されており、クライアント１２とサーバ１４の間でセキュリティ・ポリシ・プロトコルを強化する。例えば、認可プロキシ・システム１８は、認可されたアクション（以下で定義）のみが、クライアント１２とサーバ１４との間で送信された要求、メッセージ、データ、情報内容、アプリケーション、および他の情報などの内部において実施されることを保証する。認可プロキシ・システム１８は、クライアント１２とサーバ１４との間で送信された要求、メッセージ、データ、情報内容、およびアプリケーションの内部にあるコマンド、フィールド、ユーザ選択可能入力オプション、ＨＴＴＰ要求など（例えば、「要求されたアクション」）の実行をシミュレーションし、要求されたアクションが、許容可能なアクション（例えば、「認可されたアクション」）のセット内において定義されることを保証する。許容可能なアクションのセット内に存在しないあらゆるアクションは、アクションの標的（例えば、サーバ１４またはクライアント１２）に到達し、おそらくはそれを破壊する前に、認可プロキシ１８によって拒否される。

【００１７】

認可プロキシの実装戦略を変更することは、本発明の範囲内にあることを理解されたい。例えば、図１Ａは、サーバ１４とは分離した別個のネットワーク１０のハードウェア構成要素１８として認可プロキシを示す。図１Ｂでは、認可プロキシは、サーバ２０上で実行するアプリケーション・プログラミング論理（「プラグイン」論理２６）として実装されている。また、サーバ２０は、例えばデータ・ストア２８に含まれている、またはウェブ・ページ３０の上に含まれている、サーバ２０にあるアプリケーション・プログラム、情報内容、およびデータを、要求のあり次第クライアント１２に提供するためのプラグイン論理２２および２４を含む。本発明によれば、認可プラグイン論理２６は、送信する前に、クライアント１２に提供されるプログラム、内容、およびデータを確認するために、サーバ２０の上で実行される。

【００１８】

別法として、図１Ｃは、ネットワーク１０’上のルータまたはハブ４２に結合されたスニファ装置４０内において侵入検出システムとして動作する認可プロキ

シを示す。一般に知られているように、ハブ４２は、ネットワーク１０’内において通信を向ける。図１Ｃに示したように、ハブ４２は、ネットワーク１０’上の直接通信から、ウェブ・サーバ４４を隔離する。スニファ４０は、ウェブ・サーバ４０を標的にした送信を検出し、遮断する。スニファ４０は、遮断した送信を評価し、送信内において要求されたアクションを、例えば、記憶装置４６内に含まれている認可されたアクションのリストと比較するために、以下で詳述する方法を実行する。受容可能でない場合、スニファ４０は、メッセージを変更するか、または、ネットワーク上の他のシステムにメッセージを渡す。

【００１９】

本発明の実施形態について、ウェブをベースとするアプリケーションとして特定のアプリケーションを有するように記載しているが、本明細書に記載されているシステムおよび方法が、任意のクライアント／サーバ通信システム、特に、無認可の要求、データ、およびアプリケーションからサーバを保護するのが望ましい通信システム内において適用可能であることは、本発明の範囲内にあることを理解されたい。

【００２０】

以下で詳述するように、本発明の認可プロキシは、サーバ・システムの代わりに、クライアント・システム上で実行される論理を確認するために、２つの技術を使用する。第１の技術では、認可プロキシは、クライアント・サイドの論理の実行をシミュレーションし、各可能なコマンド、フィールド、ユーザが選択可能な入力オプション、およびＨＴＴＰ要求を呼び出すおよび／またはトリガする。その結果、許容可能なブラウザ・アクションの完全なセットが識別され、実際のクライアント・セッションからの後の要求を確認するために使用される。第２の技術では、プロキシは、実際のクライアント・セッション中に、クライアントサイドの論理の実行を追跡する。追跡の結果は、サーバ・リソースに対する要求内において、認可プロキシに送信される。応答して、認可プロキシは、クライアント・サイドの論理の実行をシミュレーションし、シミュレーション中に入力オプションまたは外部データに対する他の要求に遭遇するとき、追跡結果が使用される。成功したシミュレーションは、サーバ・リソースに対するクライアント要求

の承認となり、認可プロキシは、実際の処理のために、要求を適切なサーバに渡す。

【００２１】

２つの確認技術は、技術が未知のデータとユーザ介入事象とに関するクライアント論理の問合わせに応答する方法の点で異なっている。これらの技術については、以下で詳述する。

【００２２】

本発明の第１の態様では、認可プロキシ（例えば、プロキシ１８、「プラグイン」論理２６、およびスニファ４０）は、クライアントとサーバとの間の送信を評価する方法を呼び出し、サーバから発信されたメッセージ（例えば、ＪａｖａＳｃｒｉｐｔオブジェクト）に組み込まれているクライアント・サイドの論理をシミュレーションし、可能な要求されたアクションのリストを抽出する方法を呼び出す。クライアントが、要求を送信する場合、要求されたアクションは、認可されたアクションのリストに対して確認される。アクションが許容可能である場合、送信は、意図した標的の上に渡され、したがって、意図したアプリケーションに矛盾しない要求のみが実施される。

【００２３】

図２は、データ処理システムのクライアントとサーバとの間における伝送を確認するためのプロセス（例えば、前述した評価プロセス、シミュレートするプロセス、および抽出プロセス）を示す流れ図である。例えば、図２は、ウェブ・サーバからクライアント（例えば、図１のサーバ１４およびクライアント１２）への伝送（例えば、ＨＴＭＬウェブ・ページ１０２）を描いている。ブロック１００で、サーバ１４が、ＨＴＭＬウェブ・ページ１０２をクライアント１２に伝送する。伝送は、図２において線１０４Ａおよび１０４Ｂで表されている。図１Ａ～１Ｃおよび図２に示すとおり、クライアント１２に対するどの伝送も、それぞれ、認可プロキシ１８、２６、４０によって代行受信される。これに応答して、認可プロキシは、確認プロセス（その詳細を破線１０６内に示す）を呼び出す。

【００２４】

ブロック１０８で、評価プロセスが呼び出される。評価プロセスには、例えば

、ＨＴＭＬページ１０２を構文解析して、内容およびＨＴＭＬタグ、ならびにその中に組み込まれたクライアント側論理（例えば、ＪａｖａＳｃｒｉｐｔコード）を識別することが含まれる。ＨＴＭＬページ１０２中に存在する構成要素が識別されると、認可プロキシは、ページ１０２の構成要素の実行をシミュレートして（例えば、シミュレートされたブラウザ環境）、可能なすべての要求された処理を伝送の中で識別することができ、認可された処理のリストに提供できるようにする。

【００２５】

シミュレートされたブラウザ環境には、シミュレートされたブラウザのＤｏｃｕｍｅｎｔ　Ｏｂｊｅｃｔ　Ｍｏｄｅｌ（「DOM」）およびブラウザ・オブジェクトが再現される、それぞれ、ブロック１１０および１１２、ＪａｖａＳｃｒｉｐｔランタイム環境が含まれる。ＪａｖａＳｃｒｉｐｔ標準オブジェクトのいくつかは、以下にさらに詳細に説明するとおり、置き換える必要がある。認可プロキシが、シミュレートされた環境内でＨＴＭＬページ１０２の構成要素を実行する。環境内のフックが、あらゆるトリガされたブラウザ処理、例えば、ＷＥＢサーバからドキュメントを検索する処理、またはＷＥＢサーバに書式をサブミットする処理を認可プロキシに知らせる。シミュレーションが完了すると、これらの処理は、以下に説明するとおり、クライアントの実際の要求（ブロック１２０における）に対してマッチされた認可された処理のリストを補足する。

【００２６】

前述したとおり、オブジェクトのいくつかおよびそのメソッドは、クライアント・ブラウザ環境を表すようにシミュレートされる必要がある。その第１に来るのは、ＨＴＭＬドキュメント１０２から作成されるＤＯＭである。さらに、Ｎａｖｉｇａｔｏｒなどのブラウザ・オブジェクトが存在し、このオブジェクトは、Ｎｅｔｓｃａｐｅにおいて、ブラウザ・コンテキストおよぶブラウザ・ウインドウに関する情報を提供する。列挙エンジン（ブロック１１４における）が、ＤＯＭ構成要素およびＪａｖａＳｃｒｉｐｔ構成要素のシミュレーションを調整する。また、時間オブジェクト、ランダム関数などのオブジェクトも、列挙エンジンによってシミュレートされる。これらのオブジェクトすべては、スクリプトがク

ライアントの環境内、例えば、クライアント・ブラウザ１２２上でそれらにアクセスした場合、スクリプトが獲得するものと整合性を有していなければならない。例えば、ＤＯＭのテキスト領域は、ユーザがブラウザ１２２に入力した値を返さなければならず、ランダム関数は、クライアント・スクリプトがユーザのブラウザ１２２内で獲得したのと同じ値を返さなければならず、またｎａｖｉｇａｔｏｒ．ｕｓｅｒＡｇｅｎｔは、クライアント・ブラウザ１２２の名前を戻さなければならない。

【００２７】

理解されるとおり、ブラウザ・オブジェクトに入力されるデータのいくつかは、エンベロープ・データから推論され、例えば、ｎａｖｉｇａｔｏｒ．ｕｓｅｒＡｇｅｎｔは、クライアント１２から送信されたＨＴＴＰヘッダから獲得することができる（例えば、ＨＴＭＬページ１０２のオリジナルの要求の中で）。ただし、いくつかのデータは、クライアント側で辿られる実際のシナリオに依存する。ＪａｖａＳｃｒｉｐｔ中に存在するイベント・ハンドラ（例えば、ユーザの行動に応答して実行されるコード）にさらなる複雑さが導入される。ＪａｖａＳｃｒｉｐｔのＨＴＭＬスクリプト・タグは、ロードされると実行されるが、ある種のコード（例えば、イベント・ハンドラ）は、ユーザの介入があったときにだけ実行される。例えば、ＨＴＭＬ言語の入力コントロールの多くに関するｏｎＣｌｉｃｋイベント・ハンドラが存在する。対応するイベント・ハンドラは、ユーザがコントロールを選択した（例えば、「クリックした」）ときだけ実行される。

【００２８】

列挙エンジン（ブロック１１４）が、カバレッジ・メソッドを行う。例えば、エンジンは、ＨＴＭＬページ１０２中のすべのてイベントをトリガすることにより、またユーザによる可能な様々な入力に対する複数のシミュレートされた実行を行うことにより、ブラウザ１２２による可能なすべての処理をカバーするものと想定する。理解されるとおり、この環境の実行時間は、多くの入力コントロールに依存するスクリプトの場合、指数関数的に増大する。また、この環境内でランダム関数を扱うのが困難である可能性がある。シミュレートされた実行は、その実行がユーザ入力を要求するまで進行する。要求の時点で、様々な可能な入力

値でさらなる実行を記録し、シミュレートするのにユーザの行動が必要とされる。シミュレーション・メソッドのこの態様は、図３を参照して以下により詳細に述べる。

【００２９】

シミュレーションが完了すると、サーバから発信されたＨＴＭＬページ（および関連するクライアント側論理）と整合性を有する可能なすべてのブラウザ処理のリストが提供される。これらの可能な処理は、ブロック１１８で集約され、「正規の処理」として識別される。正規の処理は、認可プロキシによって利用されて、クライアント・ブラウザ１２２におけるＨＴＭＬページ１０２の特定の実行から受信される要求の処理を確認する。ブロック１２０で、クライアントの要求の処理が認可プロキシに戻され、認可プロキシによってＨＴＭＬページ１０２のシミュレーションから生成された正規の処理のリスト１１８と比較される。したがって、認可プロキシは、正規の処理のリスト１１８中にその要求および有効なユーザ入力が存在するか、存在しないかに応じてクライアントの要求を受け入れるか、または拒否する（例えば、対応する正規の処理が認可プロキシによって識別されなかった場合、要求の処理は拒否される）。正規の処理のリスト１１８は、クライアント・ブラウザ１２２からの要求の処理の受信における遅延に対応するように記憶するのが可能である（例えば、データ・ストア３０（図１Ｂ）または４６（図１Ｃ）に）ことを理解されたい。

【００３０】

図３を参照して、以下に、ＨＴＭＬページ１０２のシミュレートされた実行のなかでユーザ入力の要求および複数の可能な入力された値を扱うオプションを説明する。前述したとおり、ブロック１５０で、認可プロキシが、ＨＴＭＬページ１０２中に含まれるブラウザ・コマンドおよびクライアント論理の実行をシミュレートする。シミュレートされた実行は、入力コントロールが現れるまで（ブロック１５５で）続くか、または他のユーザ入力の要求が行われる（ブロック１６５で）。ブロック１５５で、例えば、＜ＴＥＸＴ＞、＜ＰＡＳＳＷＯＲＤ＞&＜ＴＥＸＴＡＲＥＡ＞のＨＴＭＬタグなどのフリー・テキスト入力コマンドが、ＨＴＭＬページ１０２中で検出される。フリー・テキスト入力コマンドが現れた場

合、コントロールは、ブロック１６０に進み、クライアントから受信した入力を表す固有プレース・ホルダが割り当てられる。ブロック１６０で割り当てられるプレース・ホルダは、例えば、ユーザの行動に応答してフリー・テキスト入力コマンドから生成された入力として生成されたＵＲＬ内で後に認識されるフィールドである。この後の段階で、認可プロキシは、受信したクライアント要求とのパターン照合を行う（図２のブロック１２０）。パターン照合は、「安全な」表現だけを許可するように実施する。好ましいパターン照合の技法は、本出願の出願人に共通で譲渡された米国特許出願第０９／３４５９２０号（弁理士整理番号３２６９／６）に記載されるＨＴＭＬリンク抽出器からもたらされる。プレース・ホルダを含む文字列に基づいて分岐判定（文の場合）を認識するフックが、ＪａｖａＳｃｒｉｐｔ環境の中に組み込まれる。そのような場合、これは、入力がスクリプトの論理進行に影響を与えることを意味するので、実行が無効にされる。システムは、プレース・ホルダが、生成されたＵＲＬ中に直接に現れることにより、またはプログラムの実行フローに影響を与えることにより間接に、生成されたＵＲＬに影響を与える場合、リンクを有効にすることができる。

【００３１】

制御は、ブロック１６０からブロック１６５に進むか、またはフリー・テキスト入力コマンドが全く現れない場合、ブロック１５５からブロック１６５に進む。ブロック１６５で、認可プロキシが、いくつかの可能な入力値の１つからのユーザによるコマンド要求入力の出現を検出する。そのようなコマンドが現れた場合、コントロールは、ブロック１７０に進み、現れなければ、コントロールは、ブロック１８０に進む。ブロック１７０で、列挙された入力コントロールに対する第１の入手可能な値が割り当てられる。他の可能な正規の入力値も入手可能であるので、制御は、ブロック１７５に進み、バックトラック・ログ１１６（図２）中にエントリが行われる。このエントリは、ＨＴＭＬページ１０２中の現在の実行の相対位置（例えば、実行における深さ）、ならびに可能な正規のエントリの残りの数を記録する。ログ１１６は、制御が、前の実行のポイント（深さ）にループバックし、可能な正規の入力値の次の値を選択し、次の正規の入力値で実行を続けるのを可能にすることを理解されたい。ループバックして、ＨＴＭＬペ

ージ１０２に対するすべての入力の実行をシミュレートすることにより、可能なすべてのブラウザ・コマンドおよびそれに対するユーザ入力が、前述した可能な正規の処理のリスト１１８中に収集される。

【００３２】

提供された値をログ記録した後、ブロック１８０で実行が継続する。ブロック１８０で、シミュレータが、コードの終りを検出する。終りが検出された場合、制御は、可能なバックトラック処理のためにブロック１８５に進む。終りが現れない場合、制御は、ブロック１５０にループバックし、前述したとおり、シミュレーションが継続する。ブロック１８５で、バックトラック・ログ１１６を評価して、さらなる可能な入力値が、さらなる可能な正規の処理の検出を駆動するかどうかが判定される。エントリがバックトラック・ログ１１６中に残っている場合、深さ変数を利用して、現れた複数の入力コマンドのどれかのところに実行が再配置される。再配置された後、入力コマンドの次の値が割り当てられ、シミュレーションが、相対ロケーションから先にコードの終りまで進む。そのようなループバック実行は、バックトラック・ログ１１６中の各値が尽きるまで継続される。

【００３３】

本発明者は、ＪａｖａＳｃｒｉｐｔ環境におけるバックトラッキングの一実施形態は、バックトラッキング・ポイントまで同一の値で実行を再開することであるのを確認した。別の正規の入力がこの時点で提供され、ログ記録される。このプロセスは、列挙されるすべての値が処理されるまで繰り返される。

【００３４】

前述したとおり、シミュレーション・プロセス中の可能なすべての入力およびイベントによってトリガされるすべてのブラウザ処理を集約することにより、認可プロキシは、ＷＥＢサーバで発信されるスクリプトと整合性を有する、クライアントからの実質的にすべてではないにしても、ほとんどの可能な要求のリストを獲得する。いくつかの実施形態では、このシステムは、以下に説明する追跡実施形態に対する補助として使用されて、より良好なパフォーマンスを得る。

【００３５】

以下に、本発明のシミュレーション実施形態の例を説明する。
ＨＴＭＬページ（例えば、ＨＴＭＬページ１０２）が、３つの入力コントロールを含む。
・２つの国、米国&イスラエルのオプション・リスト（もちろん、完全なリストが可能である）
・性別、男性＆女性のラジオ選択
・ユーザの名前に関するテキスト入力

【００３６】

ＨＴＭＬ中に組み込まれたＪａｖａＳｃｒｉｐｔが、以下のパターンに従ってＵＲＬを構成する。
ｈｔｔｐ：／／ｓｉｔｅ．ｃｏｕｎｔｒｙ．ｐｅｒｆｅｃｔｏｔｅｃｈ．ｃｏｍ／ｇｅｎｄｅｒ／ｎａｍｅ．ｈｔｍｌ
ここで、ｃｏｕｎｔｒｙは、国別により「ｕｓ」または「ｉｌ」であり、ｇｅｎｄｅｒは、「ｂｏｙｓ」または「ｇｉｒｌｓ」という値であり、またｎａｍｅは、入力テキスト・フィールドである。

【００３７】

以下が、ＨＴＭＬページ・コードである。

```
<HTML>
<HEAD>Test Page</HEAD>
<BODY>
<SCRIPT LANGUAGE=JavaScript>
function openPage ()
{
  var url = "http://site.";

  if (document.form1.country.selectedIndex==0)                    [BRANCH A]
    url += "us";
  else if (document.form1.country.selectedIndex==1)
    url += "il";
  else
    url += "intl";

  url += ".perfectotech.com/"

  if (document.form1.gender[0].checked)                           [BRANCH B]
    url += "boys";
  else
    url += "girls";

  url += "/" + document.form1.name.value + ".html";               [ACCESS C]

  var w = window.open (url);
}
</SCRIPT>
<FORM NAME=form1 ACTION=javascript:openPage ()>
Choose your country
<BR>
<SELECT NAME=country>
<OPTION>U.S.</OPTION>
<OPTION>Israel</OPTION>
</SELECT>
<BR>
Your gender is
<BR>
<INPUT TYPE=RADIO NAME=gender VALUE="male">male
<INPUT TYPE=RADIO NAME=gender VALUE="female">female
<BR>
Your name is
<INPUT TYPE=TEXT NAME=name LENGTH=10>

<BR>
<INPUT TYPE=BUTTON VALUE="Get My Page" onClick = "openPage ()">
</FORM>
</BODY>
</HTML>
```

【００３８】

前述したシミュレーション・プロセスは、以下のとおりコードを扱う。

１．ドキュメントの検査により、ＪａｖａＳｃｒｉｐｔを実行するイベントは、Ｇｅｔ　Ｍｙ　Ｐａｇｅボタンの上でクリックすることだけであることが与えられる。このイベントがトリガされる。これが、バックトラック・ログ１１６中に記録され、これがトリガするイベントはこれだけであるという注釈が付けられ、これにより、このエントリが使い果たされる。使い果たしは、後続のステップでより詳細に説明する。

２．ブランチ　Ａに到達するまでコードが実行される。

３．ｓｅｌｅｃｔｅｄＩｎｄｅｘ内で国を表す値が供給され、可能な２つの値、０（米国を表す）および１（イスラエルを表す）が存在する。第１の値が供給され、０の値がブランチ　Ａで供給されたことが記録される。

４．実行が、ブランチ　Ｂに進み、Ｇｅｎｄｅｒラジオ・ボックスのチェックされた値がチェックされる。可能な２つの値、真（ｂｏｙｓ）および偽（ｇｉｒｌｓ）が存在する。真の値が供給されて、バックトラック・ログ１１６に記録される。

５．実行が、アクセス　Ｃに到達するまで進み、テキスト・フィールドの値が必要とされる。プレース・ホルダが供給される。このプレース・ホルダは、通常の状況の下では現れる可能性のないＵＮＩＣＯＤＥ文字列である。このプレース・ホルダには、＜ＰＨｎａｍｅ＞というマークが付けられる。バックトラック・ログ１１６にこれをログ記録する必要はない。というのは、この値を対象とするバックトラッキングは、必要とされないからである。＜ＰＨｎａｍｅ＞のタイプが、最大長１０のテキストとして記録される。

６．オブジェクトｗｉｎｄｏｗのＪａｖａＳｃｒｉｐｔ関数ｏｐｅｎが、要求されたＵＲＬを記録する関数にシミュレートされた環境内でフックされる。構成されたＵＲＬは、次のようなものである。

ｈｔｔｐ：／／ｓｉｔｅ．ｕｓ．ｐｅｒｆｅｃｔｏｔｅｃｈ．ｃｏｍ／ｂｏｙｓ／＜ＰＨｎａｍｅ＞．ｈｔｍｌ．

７．ポリシー・エンフォーサが、最大長１０の任意のテキストに対して＜ＰＨｎａｍｅ＞をマッチさせるのが可能であるという注釈とともに、この新しい可能な

処理のことを通知される。

８．実行が再開する。値が次の可能な値、つまり偽で置き換えられた最も深いバックトラッキング・エントリ（ブランチ　Ｂ）を例外として、同じステップが辿られる。ブランチ　Ｂの可能なすべての値は、使い果たされているので、バックトラック・ログ１１６中で、使い果たされているというマークが付けられる。

９．したがって、「ｇｉｒｌｓ」の値がｇｅｎｄｅｒフィールド内で提供され、生成されるリンクは、ｈｔｔｐ：／／ｓｉｔｅ．ｕｓ．ｐｅｒｆｅｃｔｏｔｅｃｈ．ｃｏｍ／ｇｉｒｌｓ／＜ＰＨｎａｍｅ＞．ｈｔｍｌである。

１０．実行が再開する。今度は、ブランチ　Ｂが最も深い分岐であり、次に許可される値、つまりイスラエルを表す１が提供される。

１１．再びブランチ　Ｂに到達し、今度は、バックトラック・ログ１１６は存在せず、したがって、（再）更新されたエントリがログ記録され、第１の正規の処理、すなわち、真（ｂｏｙｓ）が提供される。

１２．実行が、同じパターンで再開し、今度、獲得されるリンクは、次のとおりである。

ｈｔｔｐ：／／ｓｉｔｅ．ｉｌ．ｐｅｒｆｅｃｔｏｔｅｃｈ．ｃｏｍ／ｂｏｙｓ／＜ＰＨｎａｍｅ＞．ｈｔｍｌ

１３．このプロセスは、すべてのバックトラック・ログが尽きるまで反復して（例えば、性別フィールドに対する「ｇｉｒｌｓ」の値を提供して、もう一度）続く。したがって、可能なさらなる実行パスは、存在しない。

【００３９】

ＪａｖａＳｃｒｉｐｔコードを検査する際、要求ｈｔｔｐ：／／ｓｉｔｅ．ｉｎｔｌ．ｐｅｆｅｃｔｏｔｅｃｈ．ｃｏｍ／．．．を生成する可能性がある理論上の実行パスが見られたが、ＤＯＭは、０または１以外の国に対するｓｅｌｅｃｔｅｄＩｎｄｅｘを提供しなかったので、このパスは辿られなかった。これは、どのようにＪａｖａＳｃｒｉｐｔコードとＤＯＭの両方が、許可された要求に影響を与えるかの例である。国に関して可能な値は２つしか存在しなかったので、第３の実行パスは、正規のパスではない。

【００４０】

本発明の第２の態様では、認可プロキシ（例えば、プロキシ１８、「プラグイン」論理２６、およびスニファ４０）が、クライアントとサーバとの間の伝送を評価するためのメソッドを呼び出す。第１に、クライアント側論理が、コードに挿入されることによってインストルメント化され、クライアント・システム１２上でクライアント側論理の実行を追跡する。クライアント・システム１２上でユーザによって実行されると、サーバ・リソースの要求が、実際の実行の結果（追跡結果）とともに認可プロキシにおいて受信される。認可プロキシは、クライアント側論理の実行をシミュレートして、入力オプションまたは外部データの他の要求がシミュレーション中に現れたとき、追跡結果が利用される。成功したシミュレーションは、サーバ・リソースのクライアント要求の承認をもたらし、認可プロキシが、実際の処理のためにその要求を適切なサーバに渡す。例えば、受入れ可能な処理を有する伝送が、意図された目標に伝えられ、これにより、意図されたアプリケーションと整合性のある要求だけが実行される。

【００４１】

図４は、データ処理システムのクライアントとサーバとの間の伝送を確認するためのプロセス（例えば、前述した追跡するプロセスおよびシミュレートするプロセス）を示す流れ図である。例えば、図４が、ＷＥＢサーバ２００からクライアント２５０への伝送（例えば、ＨＴＭＬウェブ・ページ２０２）を描いている。本発明によれば、認可プロキシ（例えば、それぞれ、図１Ａ、１Ｂ、１Ｃのプロキシ１８、２６、４０）が、伝送２０２を代行受信する。シミュレーション・プロセスに入る前に、認可プロキシは、クライアント・ブラウザ２５０上で生じる値およびイベントの追跡を可能にするコマンドを受信するための伝送を準備する。ブロック２０４で、伝送（例えば、ＨＴＭＬページ２０２）が構文解析されて、内容およびＨＴＭＬタグならびにクライアント側論理（例えば、ＪａｖａＳｃｒｉｐｔコード）を識別する。クライアント側論理は、ブロック２０６に進み、ブラウザ処理（例えば、入力値の要求、および実際の入力値、イベント等）を追跡するためのコードが追加される。次に、変更されたコード、つまりインストルメント化されたコードが、要求された伝送の中（例えば、ＨＴＭＬページ中２０２）でクライアント・ブラウザ２５０に渡される。

【００４２】

前述したとおり、インストルメント化されたコードは、ブラウザ処理を追跡して、追跡の結果を認可プロキシに戻す（さらなる処理の要求の中で、またはその要求に加えて）。クライアント・ブラウザ２５０が、サーバ・リソースを要求したとき、ブラウザ処理のトレース（例えば、クライアント・ブラウザ２５０上で行われた入力およびイベント）が、評価のために認可プロキシに戻される。この実施形態では、すべての入力を列挙する代りに（図２に概要を述べたシミュレーション・プロセスに関連して説明したとおり）、ブラウザが、その要求とともに、ブラウザ環境内で実際のユーザ・セッション中にスクリプトが獲得したすべての値のトレースを送信する。このトレースは、トレース・フィーダにおいて認可プロキシの中で保持される（ブロック２１２）。

【００４３】

追跡の結果が受信されると、認可プロキシが、ブロック２１０で、オリジナルの伝送（例えば、ＨＴＭＬページ２０２）の中のコードおよびＤＯＭ構成要素をシミュレートする。図４に示すとおり、シミュレーション・ステップに対する入力は、オリジナルのクライアント側コード（例えば、追跡サポート・コードを有さないクライアント側論理）、ＤＯＭ構成要素（ブロック２０８からの）、およびトレース・フィーダ（ブロック２１２）からの追跡結果を含む。追跡されたオブジェクトが、シミュレートされた環境内で照会されたとき（ブロック２１０で）、認可プロキシは、（トレース・フィーダを介して）クライアントから受信された追跡された値に関して検査する。また、追跡された値は、オブジェクトの論理上の内容規則に照らしても検査され、例えば、テキスト・フィールド内に許容可能な文字の最大数、または可能な２つの値（真／偽）のどちらかだけでしかないチェックボックス値、他のどの値も不正である。シミュレーションの結果は、ブロック２１４に渡される受入れ可能な、つまり認可されたブラウザアクションのリストである。ただし、この実施形態では、認可されたブラウザアクションは、第１の確認プロセスに関連して前述したとおり、クライアント・ブラウザにおける特定の実行から決定された値（トレースの中からの入力およびイベント）に基づき、可能なすべてのブラウザ要求に基づくのではない。クライアントアクシ

ョン（ブロック２５０から渡された）をブロック２１４で認可されたアクション（ブロック２１０から渡された）と比較して、特定のクライアント・ブラウザ実行（ブロック２５０）の要求の１つまたは複数のアクションが認可されるかどうかの判定を行うことができる。

【００４４】

理解されるとおり、ブラウザは、デフォルトで追跡を行うこと、および／または追跡の結果を送信することはしない。したがって、ブラウザに送信されるコード（例えば、ＨＴＭＬドキュメント２０２）が、シミュレーションに先立って、ブロック２０６で行われる前述したステップで変更されて（インストルメント化されて）、このコードが、ブラウザ環境内（ブロック２５０）のブラウザアクションのトレースを作成して戻す。コードに関する例としてのインストルメント化プロセスは、以下を含む。

・ＪａｖａＳｃｒｉｐｔコードが、ＪａｖａＳｃｒｉｐｔアセンブリにコンパイルされる。

・アセンブリが、検査され、すべてのｇｅｔ＿ｐｒｏｐｅｒｔｙ、ｓｅｔ＿ｐｒｏｐｅｒｔｙ、および関数コールにマークが付けられる

・ｇｅｔ＿ｐｒｏｐｅｒｔｙ命令が、関係のあるプロパティ名に照らして検査される。

・関数コールが、追跡された関数値（例えば、ランダム）およびアクション関数（例えば、新しいＵＲＬで新しいウインドウを開くｗｉｎｄｏｗ．ｏｐｅｎ（））に照らして検査される。

・関係のある命令のそれぞれが、オブジェクトが追跡に関係があるか否かを検査する前に関数コールが挿入される。

・ｇｅｔ＿ｐｒｏｐｅｔｙとして、追跡されるオブジェクトの値が、実行トレースに追加される。

・新しいＵＲＬをロードさせるオブジェクトのｓｅｔ＿ｐｒｏｐｅｒｔｙとして、トレースがＵＲＬに付加される。

・新しいＵＲＬをロードする関数コールとして、トレースがＵＲＬに付加される。

・アセンブラが逆コンパイルされて（組み込まれた逆コンパイラを使用して）ＪａｖａＳｃｒｉｐｔに戻る。

・変更されたＪａｖａＳｃｒｉｐｔコードが、オリジナルのＪａｖａＳｃｒｉｐｔコードの代りに送信される。

【００４５】

以下は、ブラウザに送信されるコードおよび変更されたコードの例である。

【００４６】

オリジナルのコードは、以下のとおりである。

```
<SCRIPT>
function my_send() {
  var req;
  var day = document.forms[0].elements.day.value;
  if ((day == "sunday"  ) || (day == "monday"  ) || (day == "tuesday") ||
    (day == "wednesday") || (day == "thursday") || (day == "friday" ) ||
    (day == "saturday"))
  {
     window.location.href = "new_page?" + day;
  }
}
</SCRIPT>
```

ＨＴＭＬドキュメントは、以下の書式を有する。

```
<FORM NAME="form1">
Purchase ticket for day:    
    <INPUT SIZE=20 TYPE="TEXT" NAME="day" VALUE="sunday">
<br><br>
Click to submit:    
    <INPUT TYPE="BUTTON" NAME="send" VALUE="Send"
onclick="my_send();"> <br>
</FORM>
```

変更されたコードは、以下のとおりである。

```
<SCRIPT>
as_.init (0);
function my_send() {
  var as_tmp = new Array();
  var req;
  var day = (as_tmp[2] = (as_tmp[1] = 0, as_tmp[0] = document.forms,
as_prop(as_tmp[0], as_tmp[1])).elements.day, as_prop(as_tmp[2], "value"));
  if ((day == "sunday") || (day == "monday") || (day == "tuesday") || (day ==
"wednesday") || (day == "thursday") || (day == "friday") || (day == "saturday")) {
(as_tmp[5] = "www.perfecto.com/JS?" + day, as_tmp[4] = window.location,
as_setprop(as_tmp[4], as_tmp[5], "href"));
  }
}
</SCRIPT>
```

ＨＴＭＬ形式に対する変更は、以下のとおりである。

```
<FORM NAME="form1">
Purchase ticket for day:    
    <INPUT SIZE=20 TYPE="TEXT" NAME="day" VALUE="sunday">
<br><br>
Click to submit:    
    <INPUT TYPE="BUTTON" NAME="send" VALUE="Send"
onclick="as_.init(1); my_send();"> <br>
</FORM>
```

【００４７】

変更されたコードは、ｄｏｃｕｍｅｎｔ．ｆｏｒｍ［０］のｇｅｔ＿ｐｒｏｐｅｒｔｙ上のａｓ＿．ｐｒｏｐを呼び出して、それが書式であることを検査し、次にｅｌｅｍｅｎｔｓ．ｄａｙ．ｖａｌｕｅに関して検査する。ａｓ＿．ｐｒｏｐ関数は、値が追跡されるべきかどうか（テキスト・フィールドの値の場合と同様に）を検査し、その値を追跡変数に追加する。ｓｅｔ＿ｐｒｏｐｅｒｔｙが、変更されてトレースが要求されたＵＲＬに追加される。ａｓ＿．ｉｎｉｔが、スクリプトのヘッドとイベント・ハンドラの両方において追加される。イベント・ハンドラがトリガされたとき、イベントがトレースに追加される。トレースは、認可プロキシに送信されると、前述したとおり処理される。

【００４８】

以下は、ユーザが月曜のチケットを購入するのを選択するプロセスの説明である。コードは、上記にリストしたとおりインストルメント化されている。ただし

、ユーザは、コードがインストルメント化されていない場合に見るものと同一の非常に簡単な書式を自らのブラウザ内で見る。

・チケット購入の曜日を選択しなければならないことを明記するテキスト

・「ｓｕｎｄａｙ」というデフォルト値を有する簡単なテキスト・ボックス

・要求を送信するボタン

【００４９】

ユーザが、「ｔｕｅｓｄａｙ」を入力して、送信ボタンを選択することによって要求を送信するものと想定する。実際に行われるのは、以下のとおりである。ブラウザ内で、

・送信ボタンを選択することにより、ｍｙ＿ｓｅｎｄ関数を呼び出すイベント・ハンドラが起動される。ただし、イベント・ハンドラを起動することは、追跡されるアクションであり、したがって、インストルメント化されたコード中のａｓ＿ｉｎｉｔによって実行トレースに追加される。

・ＪａｖａＳｃｒｉｐｔ関数であるｍｙ＿ｓｅｎｄが呼び出される。この関数は、曜日が、平日の１つの曜日にマッチすることを確認し、ページをマッチするページ、ｎｅｗ＿ｐａｇｅ？ｄａｙで置き換えるものとされる。

・ｍｙ＿ｓｅｎｄ値で、ｄｏｃｕｍｅｎｔ．ｆｏｒｍｓ［０］．ｅｌｅｍｅｎｔ．ｄａｙ．ｖａｌｕｅがアクセスされて、ａｓ＿ｐｒｏｐによって追跡される値として識別される。したがって、値「ｄｏｃｕｍｅｎｔ．ｆｏｒｍｓ［０］．ｅｌｅｍｅｎｔ．ｄａｙ．ｖａｌｕｅ＝ｔｕｅｓｄａｙ」が、トレースに追加される。

・コードであるｗｉｎｄｏｗ．ｌｏｃａｔｉｏｎ．ｈｒｅｆ＝「ｎｅｗ＿ｐａｇｅ？」＋ｄａｙが、インストルメント化されたコード中で実行されたとき、ａｓ＿ｓｅｔｐｒｏｐが、その変数を追跡されるイベント変数として識別する。通常、新しく設定されたＵＲＬに対して単一の要求が生成されていることになる。インストルメント化されたコードの中で、生成されたトレースを含む追加の要求がサーバに送信される。

【００５０】

この時点で、サーバ内に、要求とトレースの両方がある。

・オリジナルのコードがシミュレートされた環境内で実行される。
・トレース・フィーダが、送信ボタンに関するイベント・ハンドラの起動を識別し、イベント・ハンドラがトリガされる。
・イベント・ハンドラが、ｍｙ＿ｓｅｎｄを呼び出す
・ｍｙ＿ｓｅｎｄが、追跡されるオブジェクトである変数「ｄｏｃｕｍｅｎｔ．ｆｏｒｍｓ［０］．ｅｌｅｍｅｎｔｓ．ｄａｙ．ｖａｌｕｅ」にアクセスしようと試みる。
・これにより、トレース・フィーダが、実行トレースから値「ｔｕｅｓｄａｙ」を検索する。
・トレース・フィーダが、サイズ２０のテキスト・ボックス内の正規の入力にマッチすることを確認する。
・値が、ＪａｖａＳｃｒｉｐｔコードに提供される。
・ＪａｖａＳｃｒｉｐｔコードが、「ｔｕｅｓｄａｙ」は、実際に平日であることを確認し、ｗｉｎｄｏｗ．ｌｏｃａｔｉｏｎ．ｈｒｅｆ＝「ｎｅｗ＿ｐａｇｅ？ｔｕｅｓｄａｙ」に設定する。
・この属性を設定することにより、正しくマッチする実際の要求に対して「ｎｅｗ＿ｐａｇｅ？ｔｕｅｓｄａｙ」のシミュレートされた要求をマッチさせるイベントが生じる。

【００５１】

クライアント・システムのユーザ（例えば、ハッカー・オペレーティング・クライアント１２）が、サイズ２０のテキスト・フィールドに対する正規の入力を構成しない値（例えば、２１文字の入力）を提供しようと試みた場合、トレース・フィーダは、その値に不正な入力としてマークを付け、その実行を停止することにより、その値をブロックすることになる。ユーザが、不正な要求を提供しようと試みた場合には、シミュレートされた環境処理を実際の要求とマッチさせることに失敗する。

【００５２】

本発明を好ましい実施形態で説明し図示してきたが、当分野の技術者には明らかなとおり、本発明の趣旨および範囲を逸脱することなく、多くの変形および変

更を加えることができ、したがって、そのような変形形態および変更形態も本発明の範囲に含まれるものとするので、本発明は、以上に記載した方法または構成の正確な詳細に限定されるべきものではない。

【図面の簡単な説明】

【図１Ａ】

本発明の一実施形態により構成されかつ動作する、クライアント／サーバ・データ処理ネットワークのブロック図である。

【図１Ｂ】

本発明の他の実施形態により構成されかつ動作する、クライアント／サーバ・データ処理ネットワークのブロック図である。

【図１Ｃ】

本発明のさらに他の実施形態により構成されかつ動作する、クライアント／サーバ・データ処理ネットワークのブロック図である。

【図２】

本発明の一実施形態によるシミュレーション方法を使用して、クライアントとサーバの間で要求を確認する例示的なプロセスを示すフロー・チャートである。

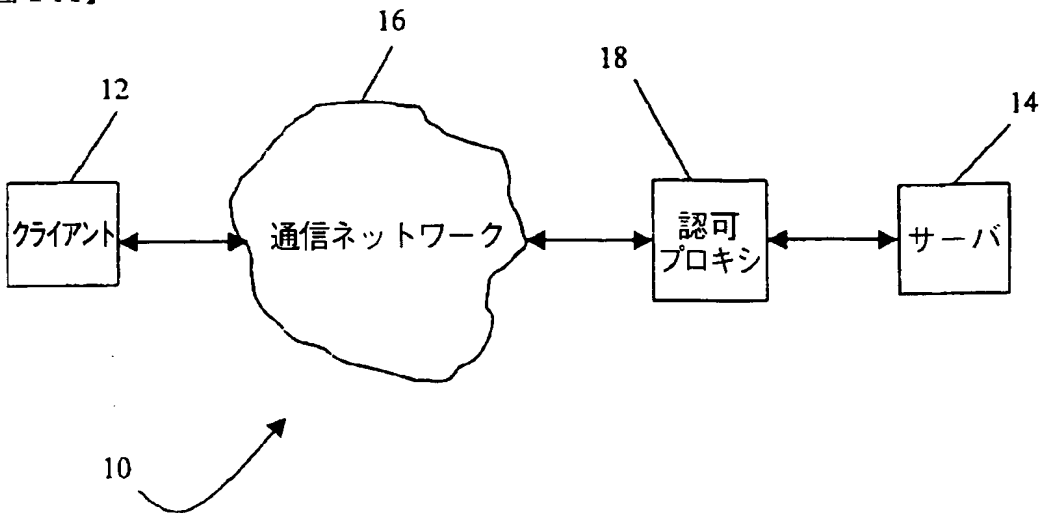
【図３】

図２のシミュレーション方法へのユーザの入力を受け取る例示的なプロセスを示すフロー・チャートである。

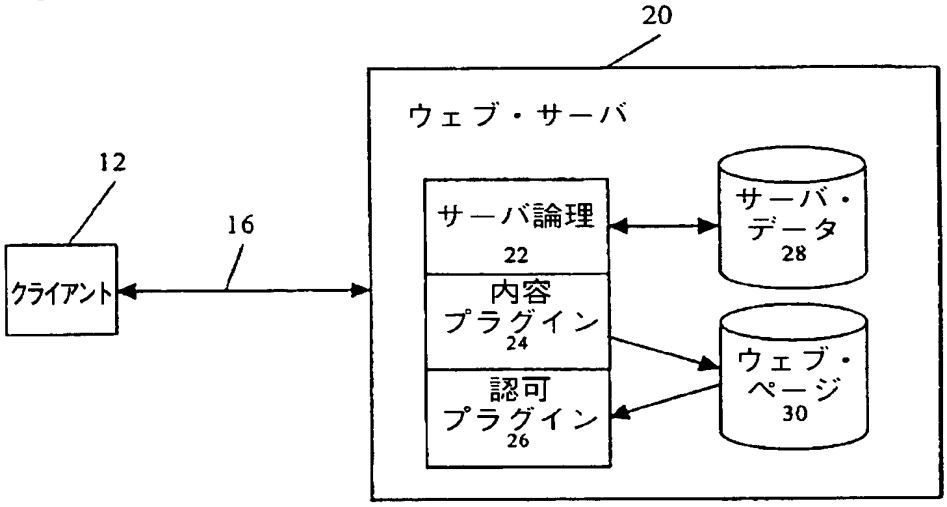
【図４】

本発明の第２実施形態による追跡方法を使用して、クライアントとサーバとの間で要求を確認する例示的なプロセスを示すフロー・チャートである。

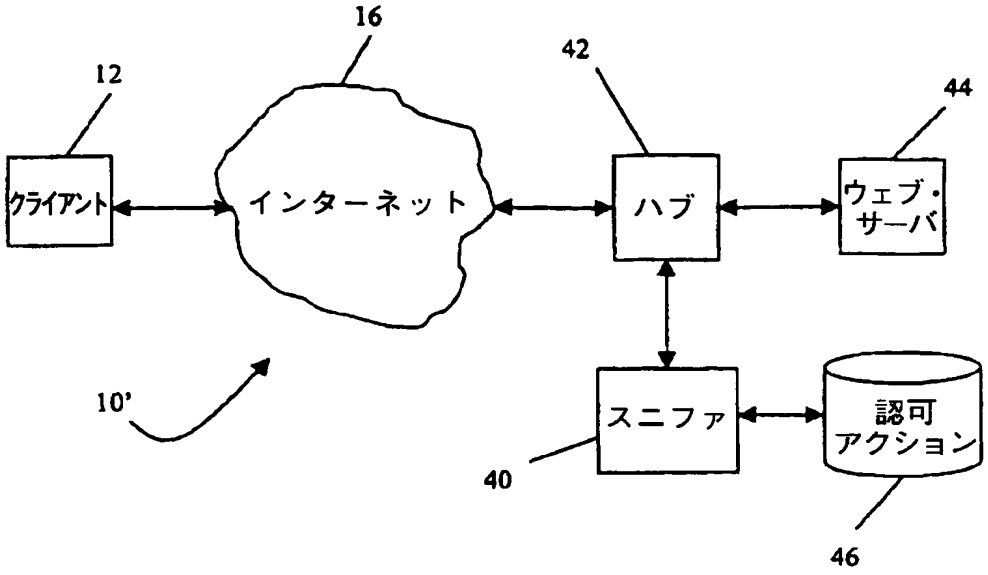
【図１Ａ】


16
18
12
14
クライアント
通信ネットワーク
認可
プロキシ
サーバ
10

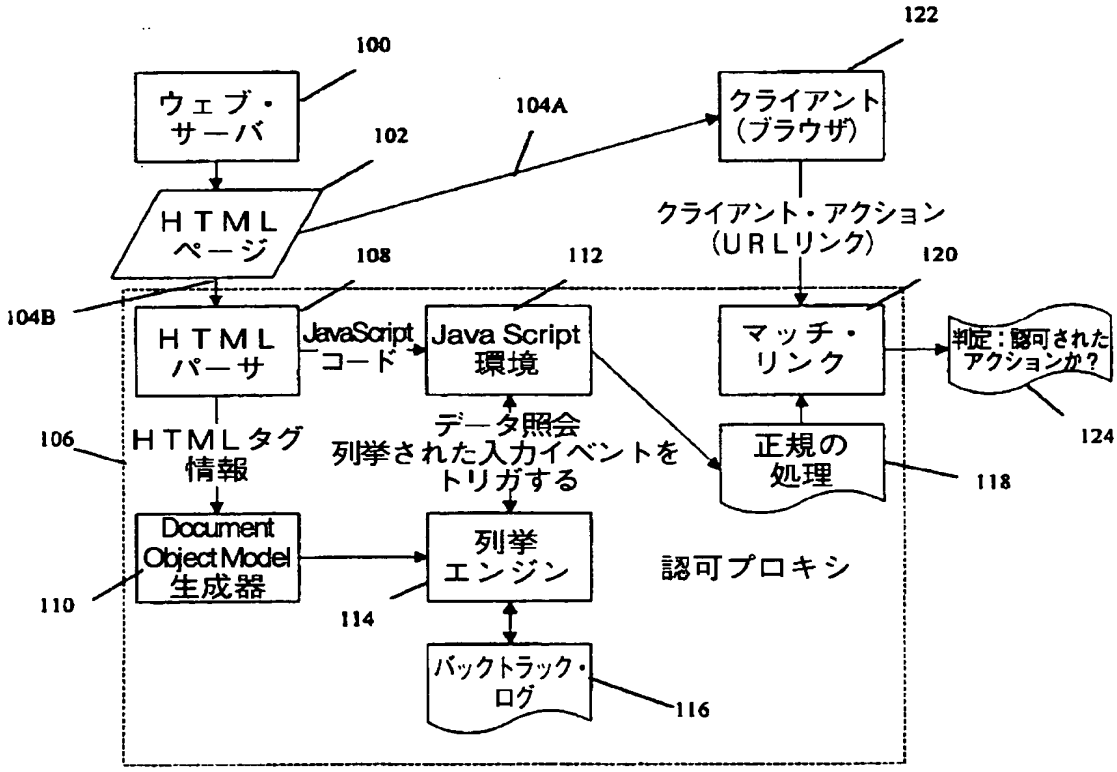
【図１Ｂ】


20
ウェブ・サーバ
12
16
クライアント
サーバ論理
22
内容
プラグイン
24
認可
プラグイン
26
サーバ・
データ
28
ウェブ・
ページ
30

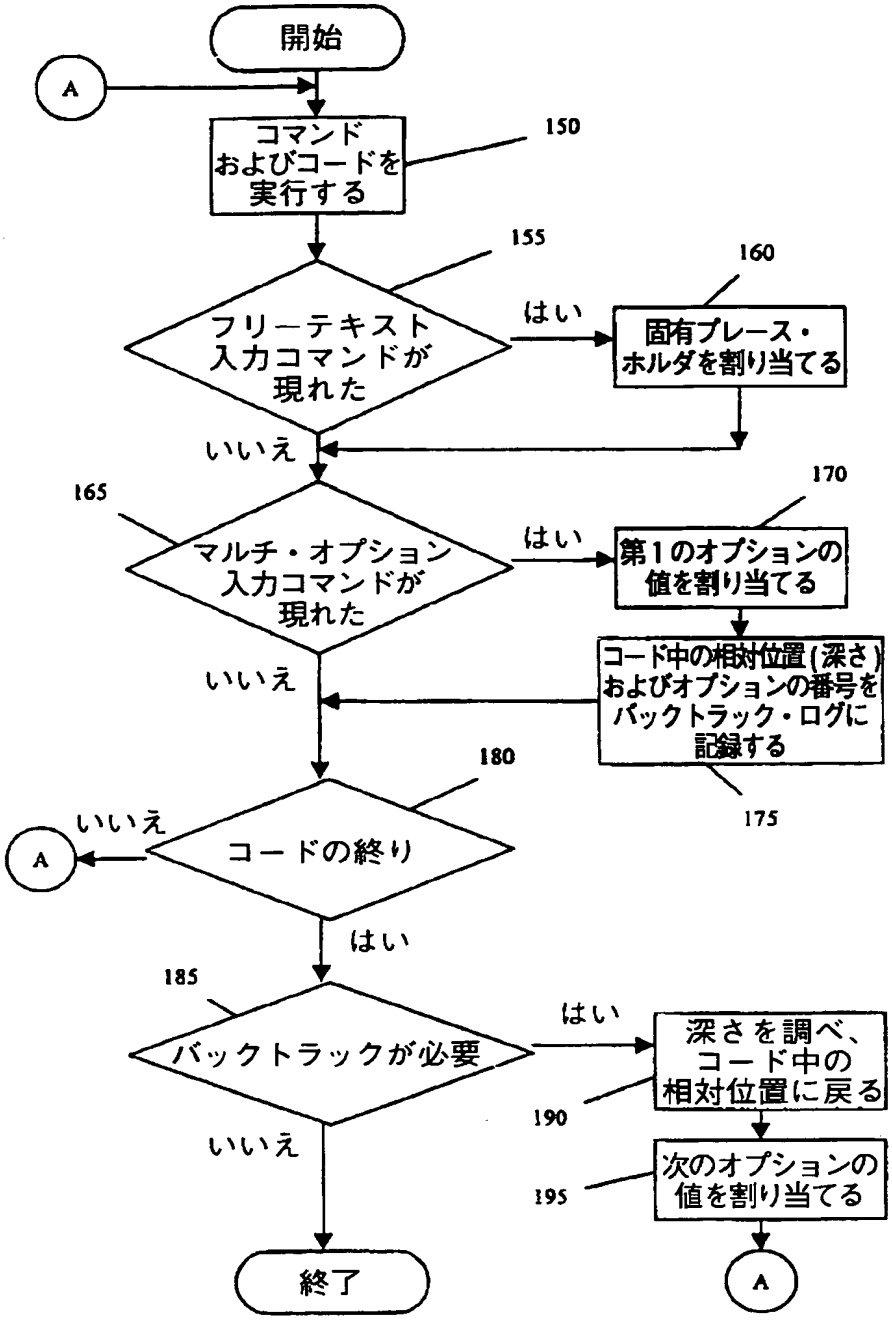
【図１Ｃ】


16
12
42
44
クライアント
インターネット
ハブ
ウェブ・サーバ
10'
スニファ
認可アクション
40
46

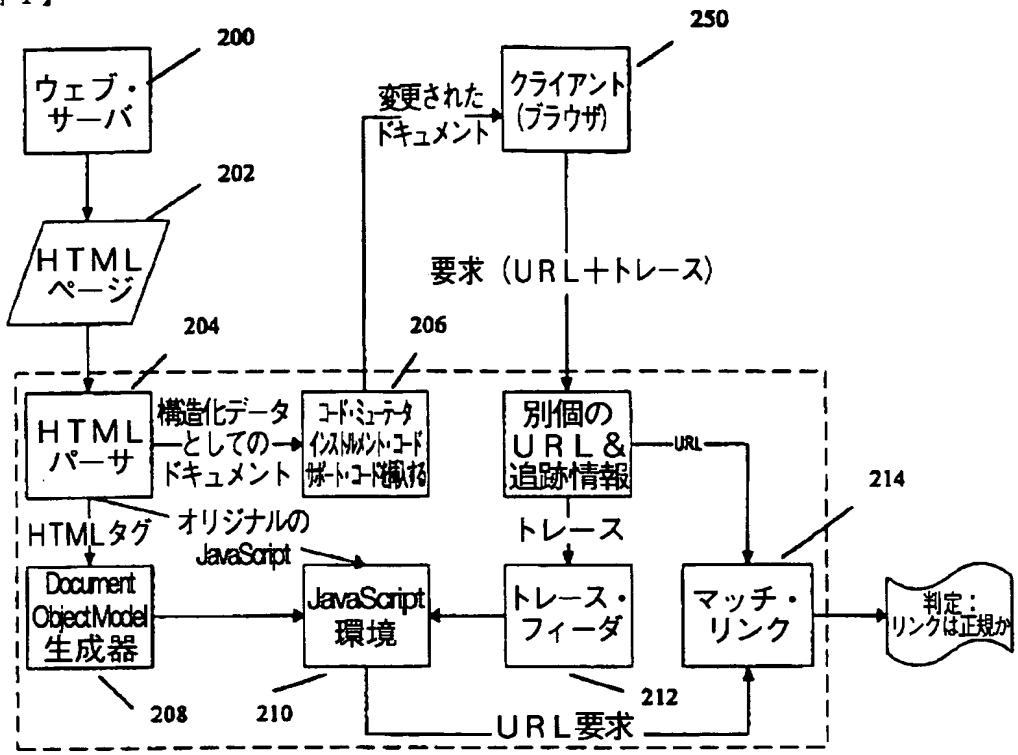
【図２】


122
100
ウェブ・サーバ
104A
クライアント(ブラウザ)
102
HTMLページ
クライアント・アクション(URLリンク)
108
112
120
104B
HTMLパーサ
JavaScriptコード
Java Script環境
マッチ・リンク
判定：認可されたアクションか？
124
106
HTMLタグ情報
データ照会
列挙された入力イベントをトリガする
正規の処理
118
Document Object Model生成器
列挙エンジン
認可プロキシ
110
114
バックトラック・ログ
116

【図３】


開始
A
コマンドおよびコードを実行する
150
155
フリーテキスト入力コマンドが現れた
はい
160
固有プレース・ホルダを割り当てる
いいえ
165
マルチ・オプション入力コマンドが現れた
はい
170
第１のオプションの値を割り当てる
コード中の相対位置(深さ)およびオプションの番号をバックトラック・ログに記録する
175
いいえ
180
コードの終り
いいえ
A
はい
185
バックトラックが必要
はい
深さを調べ、コード中の相対位置に戻る
190
次のオプションの値を割り当てる
195
いいえ
終了
A

【図４】


200
ウェブ・サーバ
250
クライアント(ブラウザ)
変更されたドキュメント
202
HTMLページ
要求（URL＋トレース）
204
206
HTMLパーサ
構造化データとしてのドキュメント
別個のURL＆追跡情報
URL
214
HTMLタグ
オリジナルのJavaScript
トレース
Document Object Model 生成器
JavaScript環境
トレース・フィーダ
マッチ・リンク
判定：リンクは正規か
208
210
212
URL要求

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL00/00677

A. CLASSIFICATION OF SUBJECT MATTER
IPC(7) : H04L 9/00; G06F 9/44, 13/10, 13/12
US CL : 703/21, 22; 713/153, 154, 164
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 703/21, 22; 713/153, 154, 164

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EAST 1.2, ACM, IEEE

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim N
X	US 5,623,601 A (VU) 22 April 1997 (22.04.1997), Abstract, 1-7b, Background of the Invention, column 3, lines 65 et seq.	1-17
X	US 5,908,469 A (BOLTZ et al) 1 June 1999 (01.06.1999), Abstract, Figures 1A-3, column 2, lines 13 et seq.	1-17
X	US 5,870,544 A (CURTIS) 09 February 1999 (09.02.1999), Abstract, Figures 3-4, Background of the Invention, column 4, lines 13 et seq.	1-17
X	US 5,347,578 A (DUXBURY) 13 September 1994 (13.09.1994), Abstract, Figures 1-3, column 1, lines 54 et seq.	1-17
X	US 5,611,048 A (JACOBS et al) 11 March 1997 (11.03.1997), Abstract, Figures 3-9, column 2, lines 65 et seq.	1-17
X	US 5,559,800 A (MOUSSEAU et al) 24 September 1996 (24.09.1996), Abstract, Figures 2-16B, column 5, lines 7 et seq.	1-17
X	SONG W. et al. Design and Implementation of a Security Management System. IEEE 1995, especially page 262, section entitled "Simulator".	1-17
X	PETERSON, K.L. IDA - Intrusion Detection Alert, IEEE 1992, pages 306-311.	1-17

☒ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
09 April 2001 (09.04.2001)	30 APR 2001
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. 703-305-3230	Authorized officer KEVIN TESKA Telephone No. 703-305-9704

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL00/00677

C (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LIN J. et al. Abstraction-Based Misuse Detection:High Level Specifications and Adaptable Strategies, IEEE 1998, Abstract, Figures 1 and 2, entire document.	1-17
X	SANDHU R.S. et al. Role-Based Access Control: A Multi-Dimensional View. IEEE 1994, Abstract, pages 54-60.	1-17
X	KOGAN B. et al. An Audit Model for Object Oriented Databases, IEEE 1991, pages 90-96.	1-17
X	FREEMAN J. et al. Developing Secure Systems: Issues and Solutions, IEEE 1988, pages 183-189.	1-17
X	BIEBER P. Formal Techniques for an ITSEC-E4 Secure Gateway, IEEE 1996, pages 236-244.	1-17

フロントページの続き

(81)指定国 EP(AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG), AP(GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW

(72)発明者 ラーナン、ギル
イスラエル国 ゾラン、ハハダリム ストリート 19
(72)発明者 レシェフ、エラン
アメリカ合衆国 カリフォルニア、サニーヴェイル、 オールド サンフランシスコ ロード 718

Fターム(参考) 5B085 AE00 BA06 BG07